ECCM-2001

European Conference on
Computational Mechanics

June 26-29, 2001
Cracow, Poland

NEW INCLUSION FUNCTIONS IN INTERVAL GLOBAL OPTIMIZATION OF ENGINEERING STRUCTURES

Andrzej Pownuk

Chair of Theoretical Mechanics, Poland
Silesian University of Technology
e–mail: pownuk@zeus.polsl.gliwice.pl
URL: http://zeus.polsl.gliwice.pl/~pownuk

Key words: optimisation, interval arithmetic, uncertain parameters, fuzzy sets

---

Abstract. *Existing optimization methods usually are not reliable or cannot use the nondifferentiable, not continuous objective functions or constraints. An interval global optimization method is very stable and robust, universally applicable and fully reliable. The interval algorithm guarantees that all stationary global solutions have been found. The convergence of this method depends on choice of a good inclusion function. In this paper a method of constructing the inclusion function is presented. This method is based on special monotonicity tests. The algorithm is applied to optimization of mechanical systems.*

---

1 Introduction

Algorithms for solving global optimization problems can be classified into heuristic methods that can find the global optimum only with high probability, and methods that guarantee to find a global optimum with some accuracy. The most important class of methods of the second type are branch and bound methods [1, 2, 6]. They originate from combinatorial optimization, where global optima are also wanted but the variables are discrete and take several values only. Branch and bound methods guarantee to find a global optimum with a desired accuracy after a predictable (though often exponential) number of steps. The basic idea is that the configuration space is split repeatedly by branching into smaller and smaller parts. This is not done homogenously, but instead some parts are preferred and others are eliminated. The details depend on bounding procedures. Lower bounds on the objective allow to eliminate large portions of the configuration space early in the computation so that only a (usually small) part of the branching tree has to be generated and processed. The lower bounds may be obtained by techniques of the interval analysis [2], or methods based on the knowledge of Lipshitz constants. In many engineering structures relation between the solutions and uncertain parameters is monotone. For monotone functions extreme values of the solution can be calculated using only the endpoints of given intervals [3, 4]. Monotonicity of the solution can be checked by using Taylor series or interval method [2]. Using monotonicity tests we can build a very good inclusion function for very large problems [4]. In this paper a new inclusion functions are applied in global optimisation method. This inclusion function can be applied for nonlinear problems of computational mechanics [4].

We can also accelerate the speed of calculation by using hybrid algorithms [5]. In this case first we find approximate solution using another optimisation method. Then we can check if the particular solution is a global solution of the optimisation problem by using interval global optimisation algorithms. In the first part of this algorithm we can apply any optimisation method particularly gradient method, genetic algorithms, sequential linear or quadratic programming method and many others.

2 Interval arithmetic

A real interval is a set of real numbers such that:

$$[x]=[x^-, x^+]=\{x\in \mathrm{R} : x^- \leq x \leq x^+\} \qquad (1)$$

The set of all intervals is denoted by *IR* and called a real interval space. Operations and functions on real numbers are naturally extended to interval operands according to the general formula [2]

$$[x]\oplus[y]=\{\mathrm{x}\oplus \mathrm{y} : x\in[x], y\in[y]\}, \quad \text{where:} \quad \oplus\in\{+,-,\cdot,/\}, \qquad (2)$$

$$f([x_1],...,[x_2])=\{f(x_1,...,x_\mathrm{n}) : x_1\in[x_1],\dots,x_\mathrm{n}\in[x_n]\}. \qquad (3)$$

Multidimensional intervals can be defined in the following way:

$$[\mathbf{x}] = [x_1^-, x_1^+] \times [x_2^-, x_2^+] \times \ldots \times [x_n^-, x_n^+] \in IR^n \; . \tag{4}$$

We call a function *f programmable* if *f* can be built up from arithmetic, logical and comparison operators and some collection of standard transcendental functions (like *sin, cos, power*, etc.). Particularly, taking an argument x, the function value $f(x)$, can be calculated with a finite number of operations.

Another important property of arithmetic operations on intervals is called *inclusion isotonicity*:

$$([a] \subseteq [c]) \wedge ([b] \subseteq [d]) \Rightarrow [a] \oplus [b] \subseteq [c] \oplus [d] \tag{5}$$

that is the result of straightforward calculation of interval expression will always include the proper result ($\oplus$ is any interval arithmetic operation).

A function $F : IR \rightarrow IR$ which satisfies

$$\{ f(x) : x \in [x] \} \subseteq F([x]) \quad \text{for all } [x] \in IR \tag{6}$$

will be called an inclusion function for f over $[x]$.

Let $[x] \in \mathrm{IR}$; then the natural interval extension $\hat{f}([x])$ of the programmable function f to $[x]$ is defined as that expression which is obtained from the expression $f(x)$ by replacing each occurrence of the variable x by the interval $[x]$, the arithmetic operations of IR by the corresponding interval arithmetic operations, and each occurrence of pre-declared function g by the corresponding inclusion function. It follows from the inclusion isotonicity of interval arithmetic operations that:

$$\mathrm{x} \in [x] \text{ implies } f(\mathrm{x}) \in \hat{f}([x]). \tag{7}$$

Property (7) is the key to almost all interval arithmetic applications and results and (7) is called the *fundamental property of interval arithmetic* [2].

For any bounded set of real numbers S we can define the *smallest interval enclosure* of the set

$$\mathit{hull}\ \mathrm{S} = [\mathit{inf}\ \mathrm{S}, \mathit{sup}\ \mathrm{S}] \tag{8}$$

3 Basic algorithm of interval global optimization

The interval global optimization method is based on the fundamental property of interval arithmetic (7). If the following inequality holds:

$$\hat{f}([\mathbf{x}_1])^+ < \hat{f}([\mathbf{x}_2])^- \tag{9}$$

where $[\hat{f}([\mathbf{x}])^-, \hat{f}([\mathbf{x}])^+] = \mathit{hull}\ \hat{f}([\mathbf{x}])$, $[\mathbf{x}_1],[\mathbf{x}_2] \in IR^n$, $f : R^n \supset [\mathbf{x}] \rightarrow R$, then the global minimum does not lie in the interval $[\mathbf{x}_2]$, and hence $[\mathbf{x}_2]$ can be omitted in future calculations (Fig. 1).

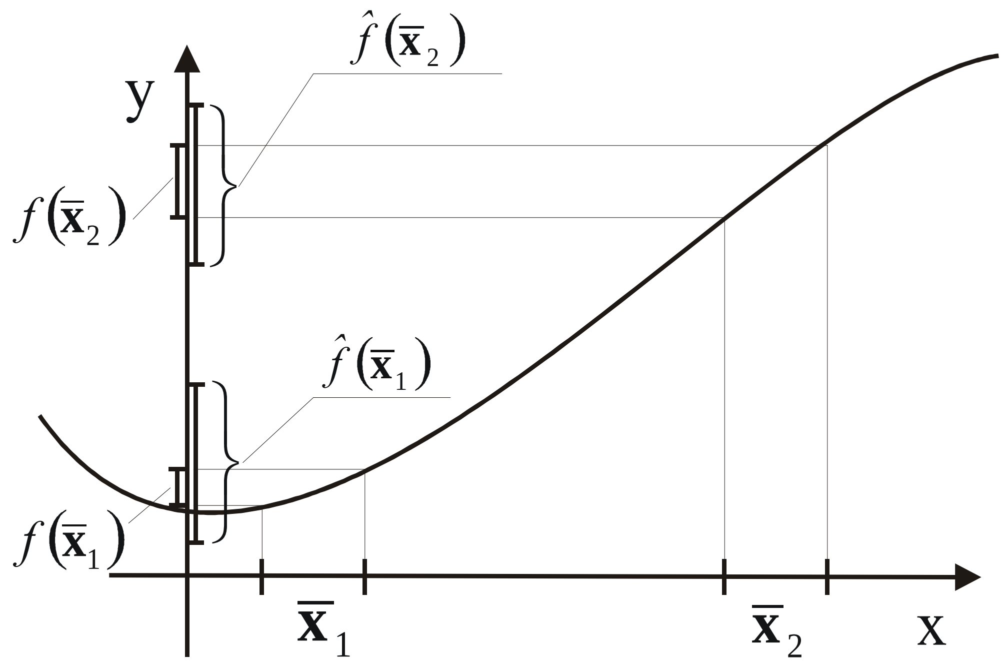

Fig. 1

For example, let $f(\mathbf{x}) = x_1^2 + x_2^2$ and $[\mathbf{x}_1] = [0,1]\times[0,1]$ and $[\mathbf{x}_2] = [2,3]\times[2,3]$, then $\hat{f}([\mathbf{x}_1]) = [0,2]$, $\hat{f}([\mathbf{x}_2]) = [8,18]$. Because $\hat{f}([\mathbf{x}_1])^+ < \hat{f}([\mathbf{x}_2])^-$ then from the fundamental property of interval arithmetic, it follows that

$$\forall \hat{x} \in [\mathbf{x}_1], \forall \breve{x} \in [\mathbf{x}_2]\ f(\hat{x}) < f(\breve{x}), \tag{10}$$

hence the global minimum of the function f does not lie in the interval $[\mathbf{x}_2]$ and $[\mathbf{x}_2]$ can be omitted in future calculations.

Let $[\mathbf{x}] \in IR^n$ be an initial interval. The basic algorithm is as follows [2]:

Step 0 Set $[\mathbf{y}] = [\mathbf{x}]$ and $y = \hat{f}([\mathbf{x}])^-$. Initialize the list L = (($[\mathbf{y}]$,y)) and the cut-off level

$z = \hat{f}([\mathbf{x}])^+$.

Step 1 Choose a coordinate direction $k \in \{1,2,...,n\}$.

Step 2 Bisect **y** in direction k: $[\mathbf{y}] = [\mathbf{v}_1] \cup [\mathbf{v}_2]$.

Step 3 Calculate $\hat{f}([\mathbf{v}_1])$ and $\hat{f}([\mathbf{v}_2])$ and set $v_i = \hat{f}([\mathbf{v}_i])^-$ for i=1,2 and

$z = \min\{z,\ \hat{f}([\mathbf{v}_1])^+,\ \hat{f}([\mathbf{v}_2])^+\}$.

Step 4 Remove ($[\mathbf{y}]$, y) from the list L.

Step 5 Cutoff test: discard the pair $([\mathbf{v}_i], v_i)$ if $v_i > z$ (where i=1, 2).

Step 6 Add any remaining pair(s) to the list L. If the list becomes empty then STOP.

Step 7 Denote the pair with the smallest second element by $([\mathbf{y}], y)$.

Step 8 If the width of $\hat{f}([\mathbf{y}])$ is less than ε, then print $\hat{f}([\mathbf{y}])$ and $[\mathbf{y}]$, STOP.

Step 9 Go to step 1.

We can divide the interval V into more than two parts. In two-dimensional case steps of the algorithm are shown in Fig. 2-4.

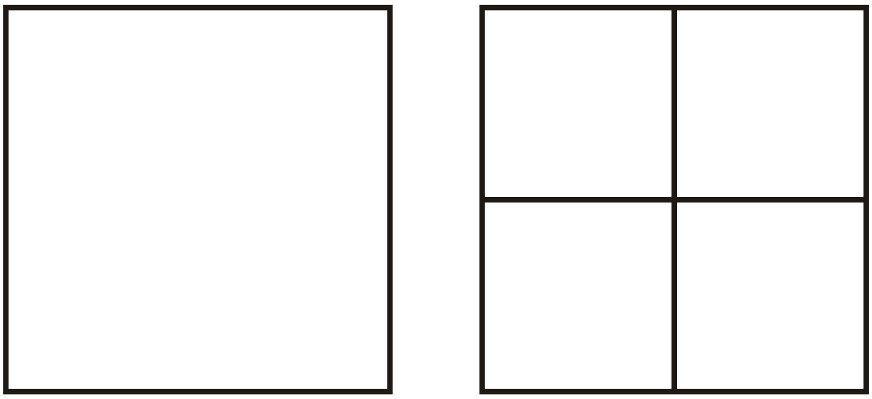

Fig. 2

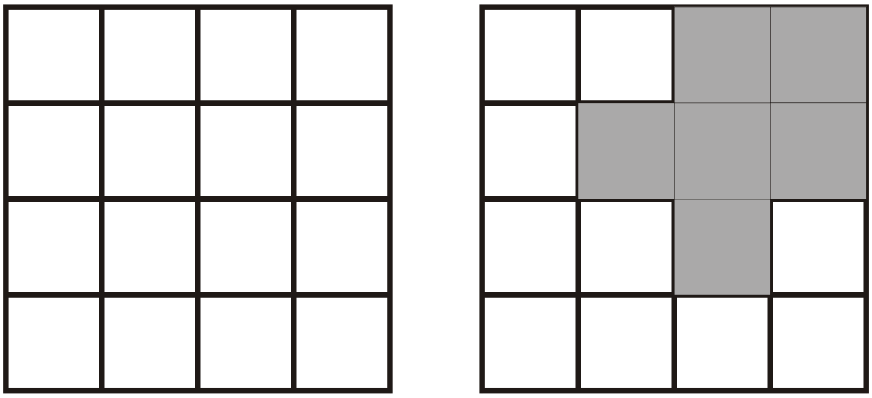

Fig. 3

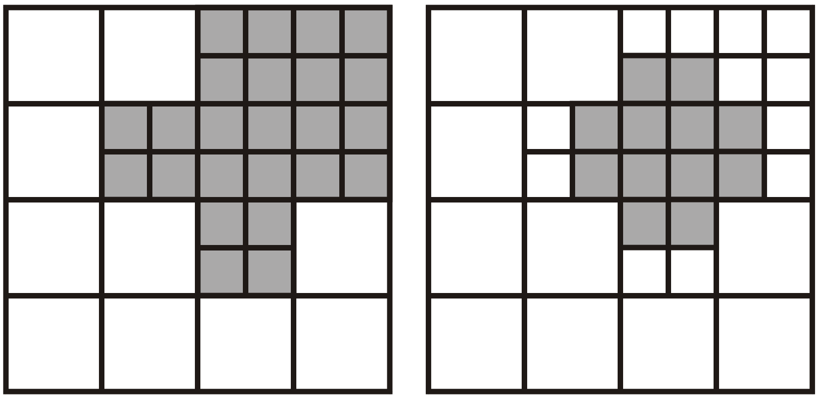

Fig. 4

4 Acceleration devices - techniques for speeding up the computation

In order to improve the quality of the interval global optimization algorithm, several special procedures should be applied.

Monotonicity test – if

$$0 \in \frac{\partial \hat{f}([x])}{\partial x_i} \text{ for some } j \in \{1,...,n\} \quad (11)$$

then there is no stationary point of f in $[x]$. In particular, $[\mathbf{x}]$ cannot contain the global minimum.

Midpoint test - if the following inequality holds (compare [9])

$$\hat{f}(\mathbf{x})^+ < \hat{f}([\mathbf{x}_2])^-, \text{ where } \mathbf{x} \in [\mathbf{x}_1] \quad (12)$$

then global minimum cannot be in the interval $[\mathbf{x}_2]$.

Finding a function value as small as possible - midpoint test gives better results if the number $f(\mathbf{x})$ is as small as possible. Minimum of the function f in $[\mathbf{x}_1]$ can be found using any local optimization method.

The interval Newton method - the interval Newton method is applied to $[\mathbf{x}]$ to determine existence or uniqueness of a zero of the gradient of the function f.

Use a good inclusion function - in a calculation it is better to use an inclusion function rather than the natural interval extension.

A nonconvexity check - if the function f has unconstrained minimum at $\mathbf{x}^*$, then f must be convex in some neighborhood of $\mathbf{x}^*$. Hence, the Hessian **H** of f must be positive semidefinite at $\mathbf{x}^*$. A necessary condition for this is that the diagonal elements H_{ii} (i=1,...,n) be nonnegative. Consider an interval $[\mathbf{x}]$. If $\hat{H}_{ii}([\mathbf{x}])^+ < 0$ for some i = 1,..., n then $H_{ii}(\mathbf{x}) < 0$ for all $\mathbf{x} \in [\mathbf{x}]$. Hence, **H** cannot be positive semidefinitive for any point in $[\mathbf{x}]$. Therefore, f cannot have a stationary minimum in $[\mathbf{x}]$ and $[\mathbf{x}]$ can be deleted.

Recursive (automatic) differentiation - using automatic differentiation we can calculate derivatives of very complicated functions.

The Fritz-John conditions - this procedure is very similar to Kuhn-Tucker conditions and we use them in constrained global optimization problems.

Use of constraints - using fundamental property of interval arithmetic (7) we can verify if in a given interval $[\mathbf{x}]$ exist any points that satisfy all constraints.

5 Point monotonicity tests

If the objective function is smooth then we can approximate it by a linear function (Fig 5).

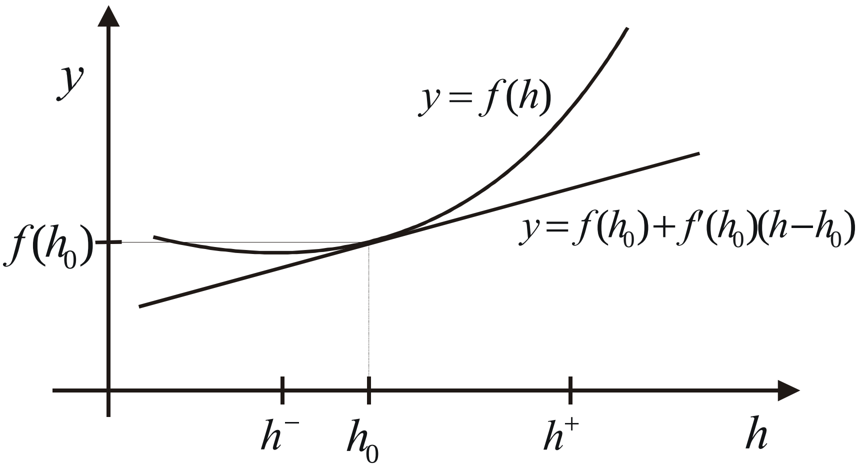

Fig. 5

$$f(\mathbf{x}) \approx f(\mathbf{x}_0) + \sum_{i=1}^{n} \frac{\partial f(\mathbf{x}_0)}{\partial x_i}(x_i - x_i^0) \tag{13}$$

where $\mathbf{x}^0 = (x_1^0, ..., x_n^0)$. We can do the same with derivatives of objective function

$$\frac{\partial f(\mathbf{x})}{\partial x_i} \approx \frac{\partial f(\mathbf{x}_0)}{\partial x_i} + \sum_{j=1}^{n} \frac{\partial^2 f(\mathbf{x}_0)}{\partial x_i \partial x_j}(x_j - x_j^0) \tag{14}$$

If the derivatives of objective function have constant sign then the function f is monotone. Using this fact and formula (14) we can check if a function is monotone. If function is monotone then extreme values of the objective function can be calculated using only the endpoints of given intervals.

$$y^- = \inf_{\mathbf{x}:\mathbf{x}\in[\mathbf{x}]} f(\mathbf{x}) \tag{15}$$

$$y^+ = \sup_{\mathbf{x}:\mathbf{x}\in[\mathbf{x}]} f(\mathbf{x}) \tag{16}$$

We can write these facts in the following form

$$[y] = [y^-, y^+] = \left[\inf_{\mathbf{x}:\mathbf{x}\in[\mathbf{x}]} f(\mathbf{x}), \sup_{\mathbf{x}:\mathbf{x}\in[\mathbf{x}]} f(\mathbf{x}) \right] = f([\mathbf{x}]) \tag{17}$$

If we know sign of the partial derivatives we can calculate the extreme values of the objective function using only the endpoints of the given interval [**x**].

$$y^- = f(x_1^{\pm},...,x_n^{\pm}) \tag{18}$$

$$y^+ = f(x_1^{\pm},...,x_n^{\pm}) \tag{19}$$

$$\text{if } \frac{\partial f([\mathbf{x}])}{\partial x_i} > 0\text{, then } y^- = f(....,x_i^-,...),\quad y^+ = f(....,x_i^+,...) \tag{20}$$

$$\text{if } \frac{\partial f([\mathbf{x}])}{\partial x_i} < 0\text{, then } y^- = f(....,x_i^+,...),\quad y^+ = f(....,x_i^-,...) \tag{21}$$

We can write these facts in the following forms:

$$y^- = f(\mathbf{x}^{-sign\left(\frac{\partial f}{\partial \mathbf{x}}\right)}),\quad y^+ = f(\mathbf{x}^{sign\left(\frac{\partial f}{\partial \mathbf{x}}\right)}) \tag{22}$$

5.1 First order monotonicity test

Let us assume that

$$\frac{\partial f_L(\mathbf{x})}{\partial x_i} = \frac{\partial f(\mathbf{x}_0)}{\partial x_i} + \sum_{j=1}^{n} \frac{\partial^2 f(\mathbf{x}_0)}{\partial x_i \partial x_j}(x_j - x_j^0) \tag{23}$$

From fundamental property of interval arithmetic arise that, if

$$0 \notin \frac{\partial \hat{f}_L([\mathbf{x}])}{\partial x_i} \tag{24}$$

then the partial derivative $\frac{\partial f(x)}{\partial x_i}$ of objective function has constant sign and the function $y = f(...,h_i,...)$ is monotone in the interval $[x_i]$. In this case we can calculate the extreme values of objective function using the formula (22).

5.2 Second order monotonicity test

If we use Taylor series we can approximate the objective function by using the following polynomial

$$f(\mathbf{x}) \approx f_{(3)}(\mathbf{x}) \overset{df}{=} f(\mathbf{x}_0) + \sum_{i=1}^{n} \frac{\partial f(\mathbf{x}_0)}{\partial x_i}(x_i - x_i^0) + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2 f(\mathbf{x}_0)}{\partial x_i \partial x_j}(x_i - x_i^0)(x_j - x_j^0) +$$

$$+\frac{1}{3!}\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} \frac{\partial^3 f(\mathbf{x}_0)}{\partial x_i \partial x_j \partial x_k}(x_i - x_i^0)(x_j - x_j^0)(x_k - x_k^0) \tag{25}$$

From this equation arise

$$\frac{\partial f_{(3)}(\mathbf{x})}{\partial x_i} = \frac{\partial f(\mathbf{x}_0)}{\partial x_i} + \sum_{j=1}^{n} \frac{\partial^2 f(\mathbf{x}_0)}{\partial x_i \partial x_j}(x_j - x_j^0) + \frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{n} \frac{\partial^3 f(\mathbf{x}_0)}{\partial x_i \partial x_j \partial x_k}(x_j - x_j^0)(x_k - x_k^0) \quad (26)$$

$$\frac{\partial^2 f_{(3)}(\mathbf{x})}{\partial x_i \partial x_j} = \frac{\partial^2 f(\mathbf{x}_0)}{\partial x_i \partial x_j} + \sum_{k=1}^{n} \frac{\partial^3 f(\mathbf{x}_0)}{\partial x_i \partial x_j \partial x_k}(x_k - x_k^0) \quad (27)$$

If

$$0 \notin \frac{\partial^2 \hat{f}_{(3)}([\mathbf{x}])}{\partial x_i \partial x_j} \quad (28)$$

then the functions $\frac{\partial f_{(3)}(\mathbf{x})}{\partial x_i}$ are monotone and the extreme values of these functions can be calculated using the sign of their derivatives.

$$\frac{\partial f_{(3)}^{-}([\mathbf{x}])}{\partial x_i} = \frac{\partial f_{(3)}\left(\mathbf{x}^{-sign\left(\frac{\partial^2 \hat{f}_{(3)}([\mathbf{x}])}{\partial x_i \partial \mathbf{x}}\right)}\right)}{\partial x_i} \quad (29)$$

$$\frac{\partial f_{(3)}^{+}([\mathbf{x}])}{\partial x_i} = \frac{\partial f_{(3)}\left(\mathbf{x}^{sign\left(\frac{\partial^2 \hat{f}_{(3)}([\mathbf{x}])}{\partial x_i \partial \mathbf{x}}\right)}\right)}{\partial x_i} \quad (30)$$

If

$$0 \notin \left[\frac{\partial f_{(3)}^{-}([\mathbf{x}])}{\partial x_i},\ \frac{\partial f_{(3)}^{+}([\mathbf{x}])}{\partial x_i}\right] = \frac{\partial \hat{f}_{(3)}([\mathbf{x}])}{\partial x_i} \quad (31)$$

then the function $f_{(3)}(\mathbf{x})$ is monotone, then we can write

$$f^{-}([\mathbf{x}]) = f\left(\mathbf{x}^{-sign\left(\frac{\partial \hat{f}_{(3)}([\mathbf{x}])}{\partial x_i}\right)}\right) \quad (32)$$

$$f^{+}([\mathbf{x}]) = f\left(\mathbf{x}^{sign\left(\frac{\partial \hat{f}_{(3)}([\mathbf{x}])}{\partial x_i}\right)}\right) \quad (33)$$

In the same way we can build higher order methods.

2 Numerical examples

Consider a problem of shape optimization of a truss structure shown in Fig. 6. The objective function will be the volume (weight) of the truss i.e.

$$\mathrm{f}(\mathbf{x}) = \sum_{i=1}^{n} A_{\mathrm{i}} \cdot L_{\mathrm{i}} = \frac{1}{\sigma_0} \sum_{\mathrm{i}=1}^{\mathrm{n}} |N_{\mathrm{i}}| \cdot L_{\mathrm{i}} \quad (1)$$

where coefficients of the vector **x** are the coordinates of the nodes of the truss.

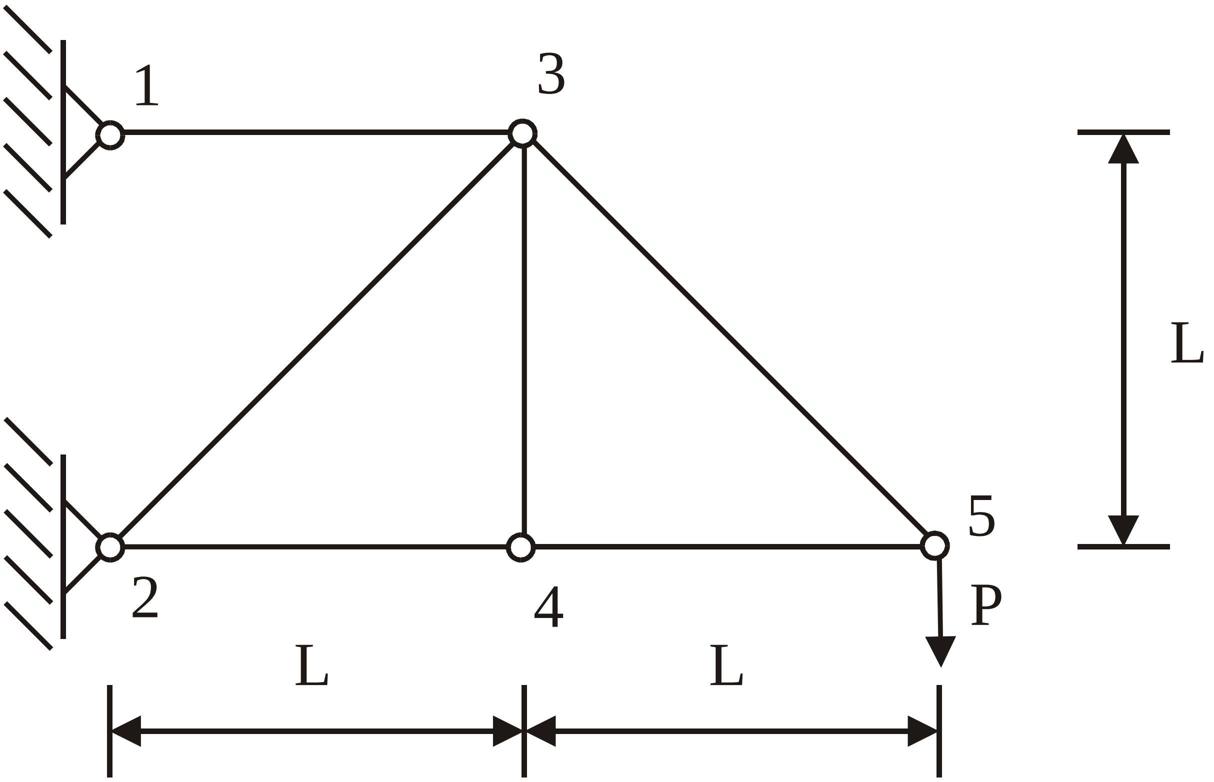

Fig. 6

The initial shape of the truss is shown in Fig. 6. Distance between the node 5 and the wall is constant and equal to 2L.

The structure must satisfy the following constraints:

- yield criterion $\frac{|N_i|}{A_i} \leq \sigma_0$,
- stability condition $|N_i| \leq \frac{\pi^2 EJ}{L^2}$.

In calculation, we assume that L=1 [m], σ_0=190 [MPa]. Numerical results are shown in Fig. 7 (P=1000 [kN]) and Fig. 8 (P=100 [kN]).

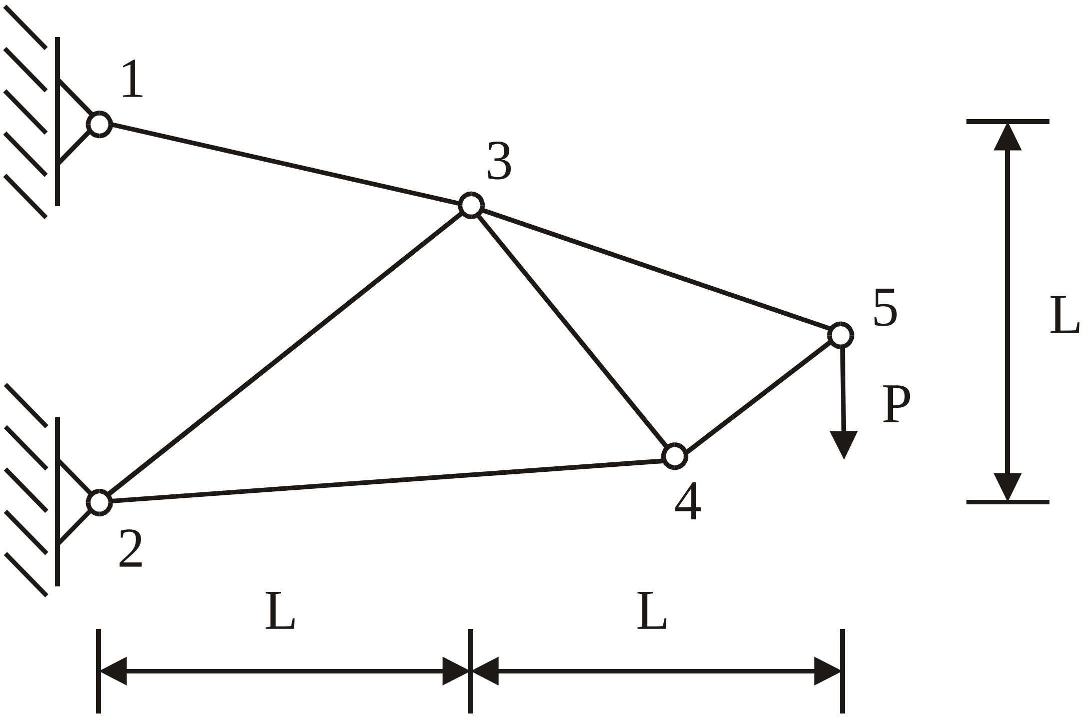

Fig. 7

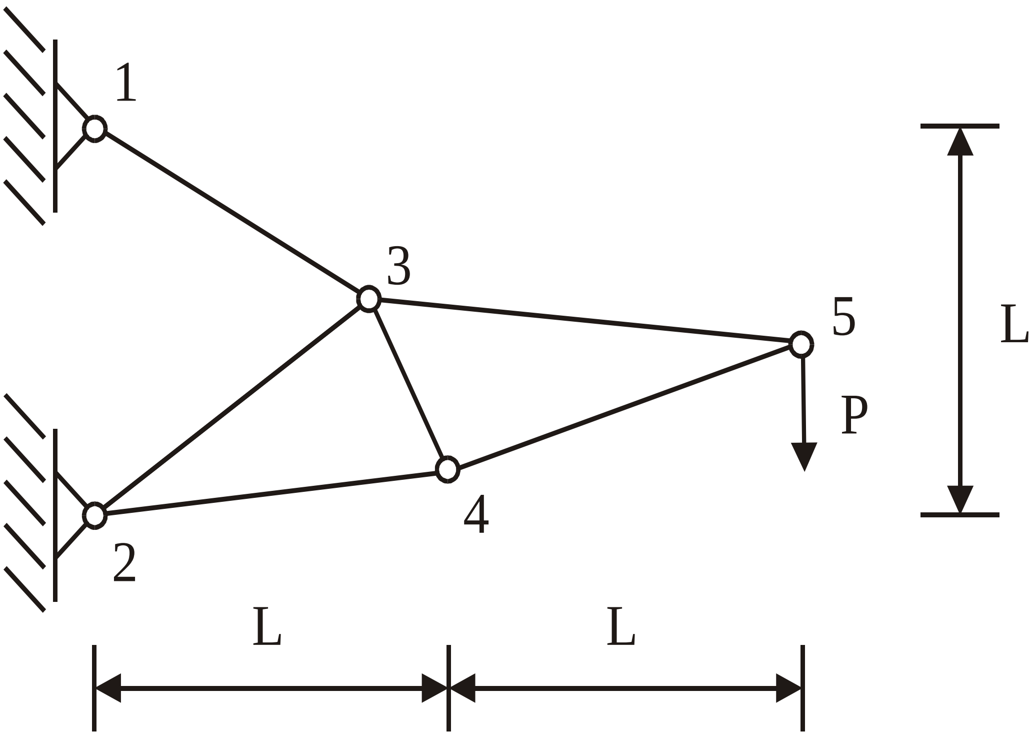

Fig. 8

3 Conclusions

The preceding results indicate that the presented algorithm is an effective and efficient method of global optimization. However, it will probably be quite slow if many local minima have values of f differing very little from the global value. The pure interval algorithm guarantees that all stationary global solutions (in the initial interval) have been found. The bounds on the solution(s) are guaranteed to be correct. Error from all sources are accounted for. The algorithm can solve the global optimization problem also when the objective function is nondifferentiable or even not continuous. If we apply the point monotonicity test we can solve complicated problems of computational mechanics. The point monotonicity tests give an exact range of the objective function over a given interval. If we apply the point monotonicity tests we cannot obtain results with guaranteed accuracy.

This work was supported by the grant No. 8T11F00615 from the Committee of Scientific research (KBN).

References

[1] Floudas Ch.A., *Deterministic Global Optimization. Theory, Methods and Applications*. Kluwer Academic Publisher, London 2000

[2] Hansen E.R., *Global Optimization Using Interval Analysis*, Marcel Dekker, New York 1992

[3] Pownuk A., Applications of sensitivity analysis for modelling of structures with uncertain parameters. *International Conference on Interval Methods in Science and Engineering Interval'2000*, Germany, Karlsruhe 2000

[4] Pownuk A., Calculation of displacement in elastic and elastic-plastic structures with interval parameters. *33 rd Solid Mechanics Conference* (*SolMech2000*), Poland, Zakopane 2000

[5] Pownuk A., Hybrid interval method for global optimisation of engineering structures. *III slovensko-posko-české sympózium "Nelineárna mechanika"*, Slovakia, Bratislava 2000

[6] Pownuk A., Optimization of mechanical structures using interval analysis. *CAMES*, Vol.7, No.4, 2000, pp.551-558